Final Project Requirements checklist.

1. Python scripts – one file (with many functions)
2. Flowcharts for (1) plus flowcharts for each of the function you have created.
3. Pseudo codes for the flowcharts.
4. Test data for the function.
5. Journal
   1. Testing, communicating with lecturer, communicating with your peer, researching, design, coding, problem solving, documentation.
6. Plan (word doc) – strategy to complete the task
7. Tracker (spreadsheet or word doc)
8. Project documentation
9. User documentation
10. Developer documentation

More details information

1. Describe a plan on implementing the game.
2. Submit document of algorithms you have created to score the accuracy of guesses.
3. Provide the following documentation.
   1. Project documentation
      1. Purpose of this project (max 150 words
      2. Scope of this project (Wordle program using Python script running on PyCharm, Spyder, VSCode, and others)
      3. Specifications (PEP8, valid words from all\_words.txt, random word from target\_list.txt, reasonable response, run in python IDE environment, properly documented, fully tested, return the guess outcome in tuple, using the numbering systems for missing, mislocation, and hit, maximum no of tries is 6, program runs correctly without crashing, guess word must be valid from all\_words.txt, no external library to be imported except ‘random’ or other library approved by your lecturer,
      4. Documentation and resources provided (eg python IDE, computer, PY4E video, PEP8 guidelines, assessment templates, target\_word.txt, all\_words.txt, advanced template, sample basic template)
      5. Project Plan (a spreadsheet with a breakdown of tasks eg., Compiling documentation, Reviewing project requirements, requirement confirmation with lecturer, Design program high level view (pseudo code), Design program high level view (flowchart), Design algorithm for the function required to complete the program (pseudo code), Design algorithm for the function required to the program (Flowchart), Create a testing plan (using a spreadsheet to list the test eg., doctest, unittest) for each algorithm,
      6. Write a journal using Microsoft word (the journal entries should include date, time, what did you do? Is there any issue? If yes how do you resolve it? What did you learn? How to improve for future task?
   2. User documentation – instruction manual on how to run the program – step by step – maximum of one page
   3. Developer documentation
      1. Program overview
      2. Algorithm (function) overview
      3. Algorithm (function) pseudo code
      4. Algorithm (function) flow chart
      5. Algorithm (function) test data
      6. Algorithm (function) coding
      7. Algorithm (function) testing
      8. Program testing
      9. Documentation using docstrings
4. Items to be submitted
   1. Project documentation (word)
   2. Python scripts
   3. All\_words.txt, target\_list.txt
   4. Flowcharts
   5. Assessment tasks
5. The word template (Journal and Project Plan)
   1. Clarified task with required personnel
   2. Planned and prioritised work
   3. Understood the applicability of specification, standards, and guidelines for the task.
   4. Discussed and demonstrated the program (F2F or online or video)
   5. Made changes in response to feedback
   6. Confirmed implementation met the specifications
   7. Obtained sign-off.
   8. WHS has been exercised.
6. Program validation
   1. Run successfully
   2. Passes the tests defined
   3. Meet client specifications
   4. Using PEP8 coding guidelines